DETAILED ACTION

Claim Status

1. Claims 4, 6, and 14-15 are pending.

Examiner's Amendment

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

3. Authorization for an examiner's amendment was given in a telephone interview with Mr. Daniel M. Northfield (reg. 50, 302) on October 9, 2008.

4. **In the claims:**

Claims 4 and 14 have been amended. Please replace all prior claims with the claims below.

Claim 4:

A computer-implemented document-searching method for searching a document having a hierarchical structure with elements separated by element identifiers, comprising the steps of:

generating an XPATH query automaton,

wherein XPATH is an XML (Extensible Markup Language) path for searching an XML document;

wherein the XPATH query automaton is a table that contains a plurality of states of backward nodes, a condition for transition, and a search state;

wherein a collection of entries of the XPATH query automaton expresses a state transition;

wherein the generating the XPATH query automaton comprises steps of:

generating and registering the state transition, wherein the generating and registering the state transition comprises:

replacing an XPATH axis including an XPATH axis in a forward direction that is exemplified as an axis child, or a descendant in an XPATH into the state transition,

replacing an XPATH axis including an XPATH axis in an opposite direction that is exemplified as axis parent, ancestor in an XPATH into the state transition,

replacing an XPATH axis including an XPATH axis in a direction of a following-sibling, or a preceding-sibling in said XPATH into the state transition,

replacement of a predicate of an XPATH into the state transition,

replacement of a logical product AND of a predicate of an XPATH into the state transition,

replacement of a logical add OR of a predicate of an XPATH into the state transition,

replacement of a logical NOT of a predicate of an XPATH into the state transition, and

wherein replacing and replacement keep an input XPATH query expression equal in terms of a search, and generates the XPATH query automaton including the plurality of states of the backward nodes, the condition for transition, and the search state, wherein the search state includes two states of said input XPATH query expression concurrently in the state transition, and wherein every axis regarding sibling relationship among nodes of the document is included in a condition for the XPATH query automaton;

a query automaton evaluator determining the state transition of a node under determination by storing a left node and a lower node in correspondence with an identified element identifier, wherein information obtained from said left node and information obtained from said lower node for the state transition is used concurrently, and evaluating said XPATH query automaton with a search result of said left node and said lower node;

storing the XPATH query automaton generated by a compiling device in a query automaton storage device;

reading out the XPATH query automaton from said query automaton storage device and storing the XPATH query automaton while reading in said document and performing a stream search with the XPATH query automaton evaluator by using states of a plurality of different types of nodes in said element identifiers included in said document and said XPATH query automaton, thereby using two inputs and a search state; and

storing an output of the query automaton evaluator in a result-storage device and thereafter outputting the stored output of the query automaton evaluator and the output of a searched node.

Claim 6:

The document-searching method according to claim 4, wherein the step of generating an XPATH query automaton comprises a step of generating an XPATH query automaton with the state transition corresponding to an initial state, a final state, and a search state registered thereon.

Claim 14:

A computer-implemented compiling method for performing a document search, comprising steps of:

generating a XPATH query automaton;

wherein XPATH is an XML (Extensible Markup Language) path for searching an XML document;

wherein the XPATH query automaton is a table that contains a plurality of states of backward nodes, a condition for transition, and a search state;

wherein a collection of entries of the XPATH query automaton expresses a state transition;

wherein the generating the XPATH query automaton comprises steps of:

generating and registering the state transition, wherein the generating and registering the state transition comprises:

replacing an XPATH axis including an XPATH axis in a forward direction that is exemplified as an axis child, or descendant in an XPATH into the state transition,

replacing an XPATH axis including an XPATH axis in an opposite direction that is exemplified as an axis parent, or ancestor in an XPATH into the state transition,

replacing an XPATH axis including an XPATH axis in a direction of a following-sibling or a preceding-sibling sibling in an XPATH into a state transition, replacement of a predicate of an XPATH into the state transition,

replacement of a logical product AND of a predicate of an XPATH into the state transition,

replacement of a logical add OR of a predicate of an XPATH into a state transition, replacement of a logical NOT of a predicate of an XPATH into the state transition, and

wherein replacing and replacement keep an input XPATH query expression equal in terms of search, and storing a plurality of states of the backward nodes in correspondence with the backward nodes into a query automaton storage device;

wherein the generated XPATH query automaton comprises registering a plurality of states of said backward nodes, a condition for transition, the search state, wherein said search state includes two states of said input XPATH query expression concurrently in the state transition, wherein every axis regarding sibling relationship among nodes of the document is included in a condition for the XPATH query automaton, and a reached state in correspondence with each other in the storage device, performing parsing of the input XPATH query expression, and identifying different types of nodes in element identifiers;

wherein the generating and registering the state transition further includes the query automaton including the plurality of states of the backward nodes, the condition for transition, and at least a search state,

a query automaton evaluator determining the state transition of a node in the document under determination by storing a left node and a lower node in correspondence with an identified element identifier, wherein information obtained from said left node and information obtained from said lower node for the state transition are used

concurrently, and evaluating said XPATH query automaton with a search result of said left node and said lower node; and

storing the output of the query automaton evaluator in a result-storage device and thereafter outputting the stored output of the query automaton evaluator and the output of a searched node.

Claim 15:

The compiling method according to claim 14, wherein said compiling method comprises a step of identifying said backward node as a left node or a lower node according to a type of said element identifier, and wherein said plurality of states are states of said left node and said lower node.

Allowable Subject Matter

5. Claims 4, 6, 14, and 15 are allowed.

6. The following is a statement of reasons for the indication of allowable subject matter.

The following is a statement of reasons for the indication of allowable subject matter. The claims are directed to a document searching method for searching a document having a hierarchical structure with elements separated by element identifiers. The current invention allows for a query automaton to determine a query expression with a special axis or predicate to be processed. Thereby, allowing for conjunctive XPATH expressions to be processed as query expressions. In doing so, the invention builds and evaluates a query automaton table based on an inputted XPATH expression and evaluates the automaton using an automaton evaluator. The present invention thus allows for every axis regarding sibling relationship among nodes to be included in the search condition for the query automaton.

With respect to independent claim 4, the prior art of record, single or in combination does not teach or fairly suggest: "wherein XPATH is an XML (Extensible Markup Language) path for searching an XML document; wherein the XPATH query automaton, is a table that contains a plurality of states of backward nodes, a condition for transition, and a search state; wherein a collection of entries of the query automaton express a state transition; wherein the generating of the XPATH query automaton comprises the steps of: generating and registering the state transition, wherein generating and registering the state transition comprises: replacing an

XPATH axis including an XPATH axis in a forward direction that is exemplified as an axis child, or descendant in an XPATH into the state transition, replacing an XPATH axis including an XPATH axis in the an opposite direction that is exemplified as an axis parent, or ancestor in an XPATH into the state transition, replacing an XPATH axis including an XPATH axis in a direction of a following-sibling or a preceding-sibling sibling in an XPATH into a state transition, replacement of a predicate of an XPATH into the state transition, replacement of a logical product AND of a predicate of an XPATH into the state transition, replacement of a logical add OR of a predicate of an XPATH into a state transition, replacement of a logical NOT of a predicate of an XPATH into the state transition, and wherein the replacing and replacement keep an input XPATH query expression equal in terms of the search, and storing a plurality of states of the backward node in correspondence with said backward node into a query automaton storage device;" and "a query automaton evaluator determining the state transition of a node under determination by storing a left node and a lower node in correspondence with an identified element identifier, wherein information obtained from said left node and information obtained from said lower node for the state transition is used concurrently, and evaluating said XPATH query automaton with a search result of said left node and said lower node" in combination with all the other claimed elements.

With respect to independent claim 14, the prior art of record, single or in combination does not teach or fairly suggest: "wherein the XPATH query automaton, is a table that contains a plurality of states of backward nodes, a condition for transition, and a search state; wherein a collection of entries of the query automaton express a state transition; wherein the generating of

the XPATH query automaton comprises the steps of: generating and registering the state transition, wherein generating and registering the state transition comprises: replacing an XPATH axis including an XPATH axis in a forward direction that is exemplified as an axis child, or descendant in an XPATH into the state transition, replacing an XPATH axis including an XPATH axis in the an opposite direction that is exemplified as an axis parent, or ancestor in an XPATH into the state transition, replacing an XPATH axis including an XPATH axis in a direction of a following-sibling or a preceding-sibling sibling in an XPATH into a state transition, replacement of a predicate of an XPATH into the state transition, replacement of a logical product AND of a predicate of an XPATH into the state transition, replacement of a logical add OR of a predicate of an XPATH into a state transition, replacement of a logical NOT of a predicate of an XPATH into the state transition, and wherein the replacing and replacement keep an input XPATH query expression equal in terms of the search, and storing a plurality of states of the backward node in correspondence with said backward node into a query automaton storage device" and "a query automaton evaluator determining the state transition of a node in the document under determination by storing a left node and a lower node in correspondence with the identified element identifier, wherein information obtained from said left node and information obtained from said lower node for the state transition is used concurrently, and evaluating said XPATH query automaton with a search result of said left node and said lower node and wherein node data is stored until then and the node data is cleared after said evaluating" in combination with all the other claimed elements.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance".

Contact Information

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to MICHAEL PHAM whose telephone number is (571)272-3924. The examiner can normally be reached on 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cottingham can be reached on 571-272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/M. P./
Examiner, Art Unit 2167

/Cam Y Truong/
Primary Examiner, Art Unit 2169